A New View of Fractal Image Compression as Convolution Transform Coding

Dietmar Saupe

D. Saupe is with the Institut für Informatik at the University of Freiburg.

Abstract

This note presents a new conceptual view of fractal image compression. It is based on image cross-correlation and the codebook coherence characteristic of fractal image compression. Our interpretation provides a close relationship to transform coding. As a first application we demonstrate a technique for accelerating fractal encoding. It is lossless, i.e., it does not sacrifice any image quality and leads to a novel application of the fast Fourier transform-based convolution.

Keywords

Fractal image compression, transform coding, convolution, cross-correlation.

I. Introduction

Fractal image compression is a recent technique for image coding envisioned by Barnsley in 1987 and first realized by Jacquin [5] in 1989. The topic is explained in the books [1], [3] and in the survey [6]. Fractal coding is fundamentally different from conventional methods. It approximates an image by the fixed point of an image operator. The specification of this operator constitutes the fractal code. The operator is a contraction in a metric space of images and, thus — by the contraction mapping principle — has a unique fixed point, which is recovered at the decoder by iteration starting out from an arbitrary initial image. Therefore, fractal image compression has also been termed *attractor coding* [8]. By now already over 200 scientific papers [10] about it have appeared.

As with any new methodology it is interesting to study interpretations from different perspectives. Several such views of fractal image compression have been considered:

1. **Iterated function systems (IFS).** Such systems are operators in metric spaces and were introduced in a mathematical paper by Hutchinson [4] in 1981, who showed that they have fractal subsets as attractors. This motivated Barnsley to search for an image compression system that models images as attractors of IFSs. Jacquin's solution of 1989 relies on a crucial modification of IFSs, namely that the mappings involved have domains that cover only part of the image. Thus, such IFSs are called *local* [1] or *partitioned* [3].
2. **Self vector quantization.** The basic fractal encoding is almost the same as a particular type of product code vector quantization (VQ), namely the so-called *mean-*

removed shape-gain vector quantization (MRSG-VQ) [8]. In that approach an image block is approximated by the sum of a DC component and a scaled copy of an image block taken from the VQ codebook. Fractal encoding differs from MRSG-VQ because the codebook is not explicitly available at the decoder but rather given implicitly in a self-referential manner.

3. **Self-quantized wavelet subtrees.** Recently it has been noticed by Davis [2] and others that in some cases fractal encoding is equal to a certain type of wavelet transform coding. The idea is to organize the (Haar) wavelet coefficients in a tree and to approximate subtrees by scaled copies of other subtrees closer to the root of the wavelet tree.

Each of these views of fractal encoding has led to a better understanding of the subject and inspired new research. For example, the similarities to VQ had already been studied by Jacquin [6] who, in fact, had imported useful classification methods developed for VQ to his fractal encoder. The relationship to wavelets opens up interesting possibilities for hybrid codes which may hold the strongest prospects for the best rate-distortion curves available with fractal techniques.

In this letter we introduce a new view of fractal image compression, namely that of *convolution transform coding.* In their book [1] Barnsley and Hurd coined the term *Fractal Transform.* However, a clear relationship to transform coding has not been apparent so far. Our interpretation fills this gap, and — similar to the other views — yields some interesting innovations and open problems related to fractal encoding.

II. Convolution Transform Coding

The principle of transform coding for grey scale images is simple. The image is subdivided into a set of blocks, e.g., squares with n pixels each. A block can be considered as a vector in n-dimensional Euclidean space with pixel intensities being Cartesian coordinates. Consider another basis of this space, e.g., the orthogonal basis $(B_1, \ldots, B_n)$ given by the discrete cosine transform (DCT) or a set of wavelet basis functions. The image blocks R are transformed and represented as vectors with respect to the new basis. In the case of orthogonal basis vectors (not necessarily of unit norm) the transform coefficients are the expressions $\langle R, B_k \rangle / \langle B_k, B_k \rangle$, where $\langle \cdot, \cdot \rangle$ denotes the inner product. They are quantized

and the subset of the most relevant ones contribute to the transform code. At the decoder the quantized coefficients are multiplied with the corresponding basis vectors and linear combinations are formed yielding approximations of image blocks which are assembled to form the decoded image. This works well when the transformation is information packing such that a few of the transform coefficients yield good approximations.

Obviously, fractal image encoding is not of this type. However, we will see that it does fit into this scheme when we allow *redundant* transformations where the transform contains many more coefficients than the input vector. Redundant transformations are not new; *frames* in wavelet theory are an example. In our case the redundant transformation is the convolution of the range block with the (downfiltered) image. To show that we need to briefly sketch the generic type of fractal image compression, however, with orthogonalization [7].

First the image is segmented into disjoint image blocks (called *ranges*). A pool of larger image blocks (called *domains*) serves as a source of blocks from which ranges can be approximated as the sum of a DC component and a scaled copy of a domain block. For a range R the domains are twice the linear size. The domains are shrunken by pixel averaging to match the range block size. This gives a pool of *codebook blocks* $D_1, \ldots, D_{N_D}$. Consider $\mathbf{1} = (1, \ldots, 1)$, the constant block with unit intensity at every pixel and orthogonalize the codebook blocks with respect to $\mathbf{1}$, yielding $\mathcal{O}D_1, \ldots, \mathcal{O}D_{N_D}$ with $\mathcal{O}D = D - \langle D, \mathbf{1}\rangle \mathbf{1}/\langle \mathbf{1}, \mathbf{1}\rangle$. For each codebook block D define the least squares approximation $s\mathcal{O}D + o\mathbf{1}$ (*collage block*) of R. The scaling coefficient s is clamped to $[-\alpha, \alpha]$ for some $0 < \alpha < 1$ to ensure convergence in the decoding and then both s and the offset parameter o are quantized. The *collage error* for range R with respect to codebook block D is $E(D, R) = \|R - (s\mathcal{O}D + o\mathbf{1})\|^2$. The block D_k with minimal collage error $E(D_k, R)$ yields the fractal code for range R consisting of (k, s, o). The decoding in a conventional fractal codec proceeds by iteration of the range approximations starting from an arbitrary initial image.

To get an interpretation in terms of transform coding we first remark that the optimal coefficients are

$$s = \frac{\langle R, \mathcal{O}D_k\rangle}{\langle \mathcal{O}D_k, \mathcal{O}D_k\rangle} = \frac{\langle \mathbf{1}, \mathbf{1}\rangle\langle R, D_k\rangle - \langle D_k, \mathbf{1}\rangle\langle R, \mathbf{1}\rangle}{\langle \mathbf{1}, \mathbf{1}\rangle\langle D_k, D_k\rangle - \langle D_k, \mathbf{1}\rangle^2}, \quad o = \frac{\langle R, \mathbf{1}\rangle}{\langle \mathbf{1}, \mathbf{1}\rangle}.$$

which are just two of the transform coefficients of R relative to the set $\{\mathbf{1}, \mathcal{O}D_1, \ldots, \mathcal{O}D_{N_D}\}$. Overall the computations in the fractal encoding are organized as follows:

- Global preprocessing: compute $\langle D, D\rangle, \langle D, \mathbf{1}\rangle$ for all codebook blocks D.
- For each range R do:
  - Local preprocessing: compute $\langle R, R\rangle, \langle R, \mathbf{1}\rangle$.
  - For all codebook blocks D do:
    - Compute $\langle D, R\rangle$ and $E(D, R)$.

The inner products $\langle R, D_i\rangle$ are the most critical quantities, since they appear in the inner loop of the computation. However, they can be derived from a convolution as follows. We define the codebook blocks D_i by *downfiltering* the image to half its resolution. Any subblock in the downfiltered image, that has the same shape as the range, may serve as a codebook block for that range. In this setting the numbers $\langle R, D_i\rangle$ are nothing but the *finite impulse response* of the downfiltered image with respect to the range. In other words, the convolution or, more precisely, the *cross-correlation* of the range R with the downfiltered image is required. This takes the calculations of $\langle D_i, R\rangle$ out of the inner loop and places them into the local preprocessing where they are obtained for *all* codebook blocks in one batch.

Therefore, the fractal code can be seen as a set of transform coefficients, just like in transform coding. One coefficient comes from the *fixed basis block* $\mathbf{1}$, the other from the convolution transform. This establishes the close relationship of fractal image coding to transform coding. Of course, the decoding still must be done iteratively.

There are many variations of fractal image compression. For example, one has considered several fixed basis blocks and linear combinations of codebook blocks for range approximations. In principle these modifications amount to saving more transform coefficients, which corresponds to the way in which standard transform coding achieves variable code rates.

As a first application of our interpretation of fractal image encoding as convolution transform coding we present a new technique for accelerating the encoding which works well for large ranges.

III. Lossless Acceleration of Fractal Image Compression by Fast Convolution

The straightforward implementation of fractal image compression by "brute force" suffers from long encoding times. A number of techniques have been developed to reduce the computational complexity [9]. Most of them are *lossy* since they trade in a speed-up for some loss in image fidelity. In contrast, with a *lossless* method the codebook block with the minimal collage error is obtained rather than an acceptable but suboptimal one. Our approach by convolution directly leads to a new lossless method, the first one that takes advantage of the overlapping of the codebook blocks. The fast convolution — based on the convolution theorem and carried out in the frequency domain — is ideally suited to exploit this sort of codebook coherence. The inner products $\langle D, R\rangle, \langle D, D\rangle$, and $\langle D, \mathbf{1}\rangle$ can all be calculated by fast convolution more efficiently when the blocks are not too small. We have implemented the fractal encoding with convolution and conducted a series of computer experiments analyzing the performance depending on image size, range size, and codebook size [11]. Our method features nearly *constant search time per range* regardless of the range size. Overall, the test shows that our approach accelerates the brute force method for ranges of size 8×8 and outperforms the other lossless methods for ranges of size 16×16 or larger.

An interesting aspect of the convolution method is that arbitrary range shapes can be accommodated simply by zero padding the pixels not in the range. Thus, the method has a strong potential in applications where an image segmentation provides large irregularly shaped ranges and a fractal code is sought. In this case other (lossy) acceleration methods are infeasible or may be more expensive.

IV. Conclusions and Future Work

We have provided a new conceptual view of fractal image compression based on the attractor codebook coherence. Just like in transform coding, the fractal code for an image block can be regarded as a set of transform coefficients, derived here from the convolution or cross-correlation of the entire image with the block. As an application we demonstrated a lossless technique for accelerating fractal encoding based on the fast

convolution transform in the frequency domain. There are some interesting open problems that might be solved based on the transform code analogy. For example, in variable rate coding it is desirable that the peak-signal-to-noise ratio tends to infinity as the compression ratio tends to 1. Disturbingly, this could not yet be achieved with the fractal technique. Is there a way to overcome this limitation borrowing techniques from transform coding?

Acknowledgments

The author thanks Hannes Hartenstein and Raouf Hamzaoui for fruitful discussions.

REFERENCES

[1] M. Barnsley and L. Hurd, *Fractal Image Compression.* Wellesley: AK Peters, 1992.

[2] G. Davis, "Self-quantized wavelet subtrees: A wavelet-based theory for fractal image compression," in: *Proceedings DCC'95 Data Compression Conference,* J. A. Storer and M. Cohn (eds.), Los Alamitos: IEEE Computer Society Press, 1995.

[3] Y. Fisher, *Fractal Image Compression — Theory and Application.* New York: Springer-Verlag, 1994.

[4] J. Hutchinson, "Fractals and Self-Similarity," *Indiana University Journal of Mathematics,* vol. 30, pp. 713–740, 1981.

[5] A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations," *IEEE Trans. Image Processing,* vol. 1, pp. 18–30, 1992.

[6] A. E. Jacquin, "Fractal image coding: A review," *Proceedings of the IEEE,* vol. 81, pp. 1451–1465, 1993.

[7] G. E. Øien and S. Lepsøy, "Fractal-based image coding with fast decoder convergence," *Signal Processing* 40, pp. 105–117, 1994.

[8] T. A. Ramstad and S. Lepsøy, "Block-based attractor coding: Potential and comparison to vector quantization," in: *Conference Record of the Twenty-Seventh Asilomar Conference on Signals, Systems and Computers,* pp. 1504–1508, 1993.

[9] D. Saupe and R. Hamzaoui, "Complexity reduction methods for fractal image compression," in: *I.M.A. Conf. Proc. on Image Processing; Mathematical Methods and Applications,* J. M. Blackledge (ed.), Oxford University Press, 1996.

[10] D. Saupe and R. Hamzaoui, "A bibliography for fractal image compression," in: *Conf. Proc. NATO ASI Fractal Image Encoding and Analysis,* Y. Fisher (ed.), New York: Springer-Verlag, 1996.

[11] D. Saupe and H. Hartenstein, "Lossless acceleration of fractal image compression by fast convolution," submitted to ICIP'96.